Cage Code 2Y865   Document 1005896   Rev -

FINAL REPORT
FOR THE
COMMON AFFORDABLE RADAR PROCESSOR (CARP)
BUILD 4.1 (OPTION 3)
PHASE III SBIR

Contract No. N00014-01-D-0225 / Delivery Order. 005

CDRL No. A013

7 December 2005

Prepared for:
Office of Naval Research
Ballston Tower One, Rm. 619-7
800 North Quincy Street
Arlington, VA 22217-5660

Prepared By:
General Dynamics Advanced Information Systems
12450 Fair Lakes Circle
Suite 800
Fairfax, VA 22033

SBIR DATA RIGHTS:
Contract No: N00014-01-D-0225
Contractor Name: General Dynamics – Advanced Information Systems
Contractor Address: 12450 Fair Lakes Circle, Ste. 500
Fairfax, VA 22033
Expiration of SBIR Data Rights Period: Five years after completion of project.

_____   _____

GDAIS Program Manager   ONR313 Program Manager

Release Date: _____   Approval Date: _____

Report Documentation Page		*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **07 DEC 2005**	2. REPORT TYPE **N/A**	3. DATES COVERED

4. TITLE AND SUBTITLE **Common Affordable Radar Processor (CARP) Build 4.1 (OPTION 3), Phase III**	5a. CONTRACT NUMBER **N00014-01-D-0225**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **General Dynamics Advanced Information Systems, 12450 Fair Lakes Circle, Suite 800, Fairfax, VA 22033**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release, distribution unlimited.**

13. SUPPLEMENTARY NOTES **The original document contains color images.**

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT **UU**	18. NUMBER OF PAGES **34**	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**			

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Table of Contents

UNCLASSIFIED

Table of Figures

1 INTRODUCTION

1.1 Background

The Office of Naval Research (ONR) Surface/Aerospace Surveillance technology program is conducting technology investigations and development of advanced radar signal and control processor technology for current and planned radar systems. The Common Affordable Radar Processor (CARP) program is investigating new processor hardware, networks, and software architectures, which can readily accept integration of new radar signal inputs, digital signal processing techniques/algorithms, system control implementations, and software system infrastructures. The project emphasizes application of state of technology Commercial-Off-The-Shelf (COTS) open architecture processing and networking solutions to Navy radar and combat system needs. The CARP program investigates the utility of emerging commercial processing solutions for application to advanced naval radar sensor systems.

1.2 Scope

The scope of this final report is to document CARP program activities completed during Build 4.1 development supported by Delivery Order 005 Option 3 of the CARP contract. The report states project development objectives followed by a detailed description of significant development stages of: requirements definition, system architectural design, development, integration and testing. The report identifies the configuration management processes performed during development and also includes a section on future plans incorporating goals and lessons learned from development to date.

UNCLASSIFIED

2   DEVELOPMENT OBJECTIVES

CARP Build 4.1 System objectives were to develop an open radar sub-array/element interface specification, and enhance and integrate open architecture compliant CARP subsystems. Specifically, the objectives of CARP Build 4.1 were:

- Definition of the subsystems and system/subsystem interfaces of a Radar Open System Architecture (ROSA), including a low power radar sub-array system, and a ROSA compliant Real-Time Array Processing Subsystem (RTAPS) encompassing Radar Control (RC), Human Machine Interface (HMI), Digital Signal Processing (DSP), Data Distribution Module (DDM) functionality, and development of an Interface Control Document (ICD) for the system
- DDM upgrade and development of a DDM Subsystem for integration in the front end of the RTAPS
- Upgrading of back-end components of the RTAPS (RC and HMI Subsystems for Build 4.1)
- Integration of the RTAPS components (DDM, RC and HMI Subsystems) consistent with the ROSA Concept identified in the Interface Control Document

2.1   Develop an Interface Control Document

A foremost design criterion of the CARP Build 4.1 system was an emphasis on the fundamental tenets of open architecture development, specifically:

1. Modular, loosely coupled design
2. System portability/hardware independence
3. utilization of competitive commercial options
4. Adherence to standards – e.g., IEEE and industry standards
5. Scalability/upgradeability
6. Open business practices – e.g., open interface specifications

To ensure the tenets are designed into the system from inception, an open system Interface Control Document was developed in Option 3 of the Delivery Order 005 contract to define system and subsystem interfaces, control schemes and message flows for the ROSA-compliant radar front-end and RTAPS.

2.2   Upgrade DDM Configuration

For Build 4.1 system processes involved with the operation of CARP DDMs were organized into a DDM Subsystem, which included the DDM Boards, associated on-board control software, and a server-based subsystem control application.

The Build 4.1 (rev. 2) DDM is a high performance embedded computing board designed to meet future radar needs in digital channel processing and digital beamforming, and represents an enhanced version of the Build 3.0 (rev.1) DDM board. It utilizes state-of-the-industry commercial components to harness technology progression and has designed-in flexibility and modularity to enable integration with multiple system implementations.

The key goals for the Build 4.1 DDM Subsystem were the following:

- DDM Subsystem operation in an environment of defined open interfaces
    - Control and radar event messages use consistent with the ICD
    - System interface and subsystem control from a COTS server-based application including -

- ▪ Subsystem and component configuration and state/mode management
- ▪ Radar Event Message handling
- ▪ Subsystem Status monitoring
- ▪ Inter-Process Communications
  - o Board control from on-board microprocessor/software
  - o Synchronized row operation across multiple boards, processing up to eight input channels
  - o Synchronized Time-of-Day across DDM boards
- Row Beamformer open-interface-consistent loopback stimulation
  - o Transmission of pre-generated receive data files to input channels
    - ▪ Internal loopback utilizing on-board SRAM to Field Programmable Gate Array (FPGA) 10 Gigabit Ethernet (10 GE) connections using Internet Protocol (IP)
    - ▪ External loopback via DDM Board 10 GE optical port transmission/reception
  - o System stimulation at 100 and 160 Mega Samples per Second (MSPS) per channel
  - o Data format consistency with ICD specified open interface
- Row Beamformer Receive Data Processing
  - o Receive digitized Intermediate Frequency (IF) data for four channels per board at a either input rate (100 or 160 MSPS) via either loopback mode
  - o Filtering and decimation of each channel to generate 80 MSPS, 50 or 40 MSPS complex signals, supporting 2:1 and 4:1 decimation and utilizing fixed filter coefficient sets per selected decimation rate
  - o Equalization of each channel with a 32-tap complex FIR filter utilizing updatable 32-bit complex coefficients
  - o Multi-board digital beamforming utilizing board Advanced Telecom Computing Architecture (ATCA) backplane interfaces
    - ▪ Generation of four azimuth beams from eight input channels using updatable 32-bit complex coefficients
  - o Row beam data output via 10 GE test ports
  - o Row beam data output over board backplane interfaces in a format specified for input to a Column Beamformer DDM board

2.3   Upgrade and Integrate Back-End Server Based Components of the RTAPS

For Build 4.1 the processes involved in the back-end operation of the CARP Radar Array Simulation Test-Bed were organized into two functional subsystems: a Radar Controller (RC) Subsystem and a Human Machine Interface (HMI) Subsystem. A Digital Signal Processing (DSP) Subsystem encompassing the digital signal processing applications utilized in Build 3 is planned for integration in a later build (see section 4, Future Build Development Plans).

The RC Subsystem provides the system with radar control functions, including system resource management and radar event scheduling. The HMI Subsystem includes the system operator interface with system control, radar event request, and system component status window interaction. Both subsystems contain upgraded/expanded versions of applications utilized in Build 3 and are ROSA-compliant.

The subsystems were planned for integration to form the Radar Array Simulation Test-Bed prior to system integration with the DDM Subsystem and the Open System Specification concept.

UNCLASSIFIED

2.4   Integrate and Test the System

Full CARP System integration with the ROSA principles identified in the Interface Control Document was performed to allow end-to-end system operation and testing and to verify performance of the integrated RTAPS. Integration also allowed analysis of system signal processing performance and placement of hooks for integration of additional system functions in the future.

UNCLASSIFIED

3   DETAILED DESCRIPTION

The CARP system provides a common processing platform for ROSA-Compliant radar systems. By providing a modular, scalable architecture, the design could be configured to meet the requirements of a multitude of radar systems. The long term focus of the project is to support Intermediate Frequency (IF) signal reception of digital packet data and transmit signal generation/control, perform signal transmission/reception, carry out digital beamforming of received signals across multiple channels (along with down conversion and channel to channel equalization), and apply additional "downstream" signal processing on the generated beam data in commercially available computer processors. Built in testability, including loop-back and self test, is also an important goal in this effort.

3.1   Requirements Definition

Build 4.1 system requirements definition began with specification of supported inputs and data rates, and involved significantly increased DDM, Test-Bed and integrated system capabilities:
* Built in ROSA compliance
* As in Build 3, DDM capabilities including FPGA-based digital down conversion, equalization filtering and digital row beamforming
* Increased maximum digital input signal bandwidth to 160+ MHz
* Increased DDM Subsystem output bandwidth to 60+ Mega Samples Per Second (MSPS)
* Server-based radar control including radar event scheduling
* An enhanced system HMI
* Control and status messages via CARP Communication Libraries and GDAIS Multipurpose Transportable Middleware (MTM)
* 1 Gigabit Ethernet (GE) and 10 GE interfaces for messaging and RF data transfer
* Time synchronization between DDM and RC subsystems to 1 microsecond
* Data insertion and extraction at all processing points
* Built in testability and loopback system stimulation

3.2   Conceptual Design Overview

3.2.1   Interface Control Document

The purpose of the ICD is to identify major subsystems, internal interfaces between the subsystems, and external interfaces to the combat system of an integrated ROSA compatible radar front end. Furthermore, it defines and describes in detail the (1) physical digital and analog signal interfaces between the major subsystems of the radar system (2) protocols on the digital communication channels, and (3) message structures of the digital interfaces for internal interfaces. In defining the subsystems and their functional allocations, attention is given not to preclude different methods of implementation of each subsystem and natural breaks in functionality that would provide as simple an interface as possible. Furthermore, in defining the interfaces, those described in the document are defined to be as "open" as possible where open is subsequently described.

3.2.2   Data Distribution Module

In the CARP approach, the DDM Baseboard is an FPGA-based processing board that can support a variety of front end input methods, with multiple 10GE inputs to provide the beamforming processing for received signals. The DDM Subsystem, as mentioned above, encompasses control processes and multiple DDM boards.

DDMs provide the capability to self-stimulate, receive, then filter, decimate to a desired bandwidth, and equalize each received channel prior to beamforming. The CARP system supports a large number of input channels by cascading the digital beamforming DDMs with high-speed connectivity between boards to support complex phased array systems in both azimuth and elevation dimensions. Final Beamformer output is transmitted via 10 Gigabit Ethernet (10 GE) interfaces. DDM Board operations are controlled by an on-board processor. DDM Subsystem-to-system interfaces are provided by a COTS server based Subsystem Scheduling application handling functionality listed in the detailed description below.

3.2.3   Build 4.1 RTAPS

The Build 4.1 RTAPS consists of three ROSA-compliant subsystems providing the system with a DDM configuration and COTS Server based HMI and Radar Control, with open interfaces as defined in the Interface Control Document.

3.2.3.1   DDM Subsystem

As mentioned, the DDM Subsystem configuration encompasses a multi-board DDM setup and a COTS server based DDM Scheduler application serving as a subsystem-system gateway.

3.2.3.2   RC Subsystem

The RC Subsystem is comprised of a COTS Server based Radar Controller Application interfaced with all other subsystems to provide system synchronization, control, status monitoring, radar event scheduling and radar event message generation. The application utilizes CARP Messaging Libraries and GDAIS MTM.

3.2.3.3   HMI Subsystem

The HMI Subsystem is comprised of a COTS Server based HMI Application interfaced with the other subsystems to provide an operator interface allowing Radar Event request processing and interaction with system control and status capabilities. The application utilizes CARP Messaging Libraries and GDAIS MTM.

3.2.3.4   System Infrastructure

The three subsystems of the Build 4.1 RTAPS are integrated with the ROSA Concept and communicated via CARP Communications Libraries using interfaces defined in the ICD over a COTS 1 GE Network.

UNCLASSIFIED

CAGE CODE	NUMBER	REV
2Y865	1005896	-

3.3 Requirements

Requirements for the Build 4.1 System were defined to meet all Build 4.1 targets and included specifications for: system interfaces and control, simulation/stimulation functionality, status monitoring, system signal performance and integrity through all signal processing stages, system testability and performance.

3.4 Modeling and Analysis

Parallel to system design efforts, processes of the CARP system were modeled and analyzed to verify algorithms and functionality and provide stimulation or other data for use in testing. These include test data generation scripts, DSP algorithm models, radar event scheduling scheme models, system sizing analysis and system timing tests. Some models and performance values utilized in analysis were carried from previous CARP Builds and Government-provided studies.

3.4.1 Matlab System Modeling

DSP models were generated for each stage of the Build 4.1 Signal Processing data path utilizing the Matlab Fixed Point Toolbox for algorithm verification, test data generation and exact-match expected results for process testing. These included:

- Fixed Point Digital Signal Simulator/Dwell Data Generator
- Fixed Point Digital Down Conversion
- Fixed Point Equalization Filtering and Coefficient Generation
- Fixed Point Row Beamforming

The addition of the Matlab fixed point toolbox allowed modeling of system components with appropriate bit widths and fixed point processing, and permitted system dynamic range analysis and exact-match comparison with DDM processing stages during testing.

The algorithms used in the Radar Controller Subsystem for dwell scheduling were modeled in Matlab for comparison with system application output for verification. These models, along with component performance analysis (see Section 3.4.2) were used to size and characterize system components.

Matlab models were also used to specify and analyze various system parameters, including coefficients for the Beam Former and Decimation and Equalization Filters.

3.4.2 Analysis

Performance analyses were conducted on system components with heavy expected loads to calculate system performance levels required to handle data rates and sizings and to define future system configurations and parameters. Along with hardware component and software process performance testing, the studies included sizing and timing of Ethernet Network technology and evaluation of COTS high density processing systems.

3.4.2.1 Ethernet Network Technology Evaluation

The CARP System has high RF data bandwidth objectives from end-to-end and requires a very high performance network architecture capable of handling both 1 GE and 10 GE traffic. Prior to initiating

system design, an analysis of COTS Network Switches was performed, with the goal of specifying a 10 GE Switch to distribute high rate signal data between the DDM Subsystem and back-end COTS servers.

Specifically the CARP System will require a cost-efficient switch solution with the following capabilities:

- Long memory buffering on 1 GE ports (this attribute determines the maximum data burst/PRI length supported by the network)
- Traffic Shaping on the 1 GE Ports
- IEEE 802.4ad Link Aggregation for redundancy
- Jumbo Frame support
- Port Mirroring
- Scalability to the desired number of sub-array channels and beams

It was determined that to process the data associated with 1-beam the architecture must have a port density of 96 1GE Ports and 4 10 GE Ports. For a 16 Beam setup those values extend to 234 1GE and 35 10GE ports. More challenging goals for the switch included:

- Sufficient buffering to support 10GE to 1GE rate differences for long PRI lengths in a fully loaded switch chassis
- High port density and small footprint
- Support of 1 GE Traffic Shaping in a fully loaded switch chassis
- Support of link aggregation for redundancy in a fully loaded chassis
- Support for port mirroring in a fully loaded chassis

Commercial configurations from 18 vendors were researched and a Force10 E1200 Chassis configuration was determined to be the best balance of cost, schedule, risk and performance for the CARP program. Further evaluation was completed at a vendor test site in Calabasas, CA with a GDAIS representative witnessing test results. Testing involved measuring switch performance using an Ixia Network Traffic Generator/Analyzer for stimulation and recording, varying input data rate, burst length, and network configuration.

The Force10 Switch was found to support the 16 Beam Performance Objectives with Link Aggregation, Traffic Shaping and Burst Buffering without issue and exceeded buffering goals for representative worst case port mapping layouts, with upgradeability to greater performance goals supported by vender future development plans. The Force10 setup currently only lacks full support for the port mirroring objective, which should be addressable with an optical tapping setup.

3.4.2.2   *COTS High Density Server Systems Evaluation*

A near future objective of CARP development is the integration of a CARP Digital Signal Processing subsystem; with the ultimate goal of implementing a ROSA-compliant open architecture DSP Radar Processing Solution. The subsystem needs to be affordable but high density, highly scalable and server based, and able to process 16 full Beams at high data rates. An evaluation of COTS high performance server systems was completed to identify a hardware configuration for an initial version of the subsystem.

Goals of the evaluation effort included:

- Performing evaluations of several COTS DSP Subsystem candidate configurations

UNCLASSIFIED

CAGE CODE	NUMBER	REV
2Y865	1005896	-

- Perform a market survey to identify alternative solutions
- Pruning of candidate solutions based on cost, performance, and functional constraints
- Run tests and benchmarking on candidate hardware
- Establish a mitigation path to meet the ultimate subsystem objectives

Key considerations to be taken into account were:

- Performance
- Processor density
- Minimizing software complexity and associated costs
- Endian environment homogenization within the system (Note DDMs currently run big endian PowerPCs)
- High density through lower power consumption and heat dissipation.
- Openness, availability and development/device support
- Footprint/form factor
- Gigabit Ethernet Connectivity
- Ease of migration to 16 beam solution (from a near term goal of 1 beam)
- Cost effectiveness when balanced with other program priorities

Four candidates were selected from market research. Evaluation of the candidates against the listed considerations resulted in the selection of an IBM eCenter JS20 Blade configuration.

The IBM JS20-based Bladecenter met all evaluation criteria. JS20 Power PC based blades are big endian and relatively low power, and density is improved by 100% over previously tested 1U server solutions. In addition, Bladecenters support standard Linux distributions, fit a standard form factor, are highly scaleable and contain sufficient external copper GE ports to handle input data rates goals (in addition to in chassis midplane and backplane communications). Initial benchmarking rates a single JS20 Blade as 52.6% faster at short length Pulse Repetition Interval (PRI) PRI Pulse Compression than a single 1U dual-Opteron server from Build 3, and 46.5% faster on long pulses.

3.5   Build 4.1 System Architecture

The Build 4.1 CARP Architecture is comprised of three ROSA-compliant integrated subsystems of the RTAPS: a DDM Subsystem, an RC Subsystem and an HMI Subsystem. Figure 1: CARP System Conceptual Design, diagrams the conceptual subsystems and interconnections, along with Digital Signal Processing (DSP) and Digital Receiver/Exciter (DREX) components planned for development in future builds, and a conceptual external Combat System.
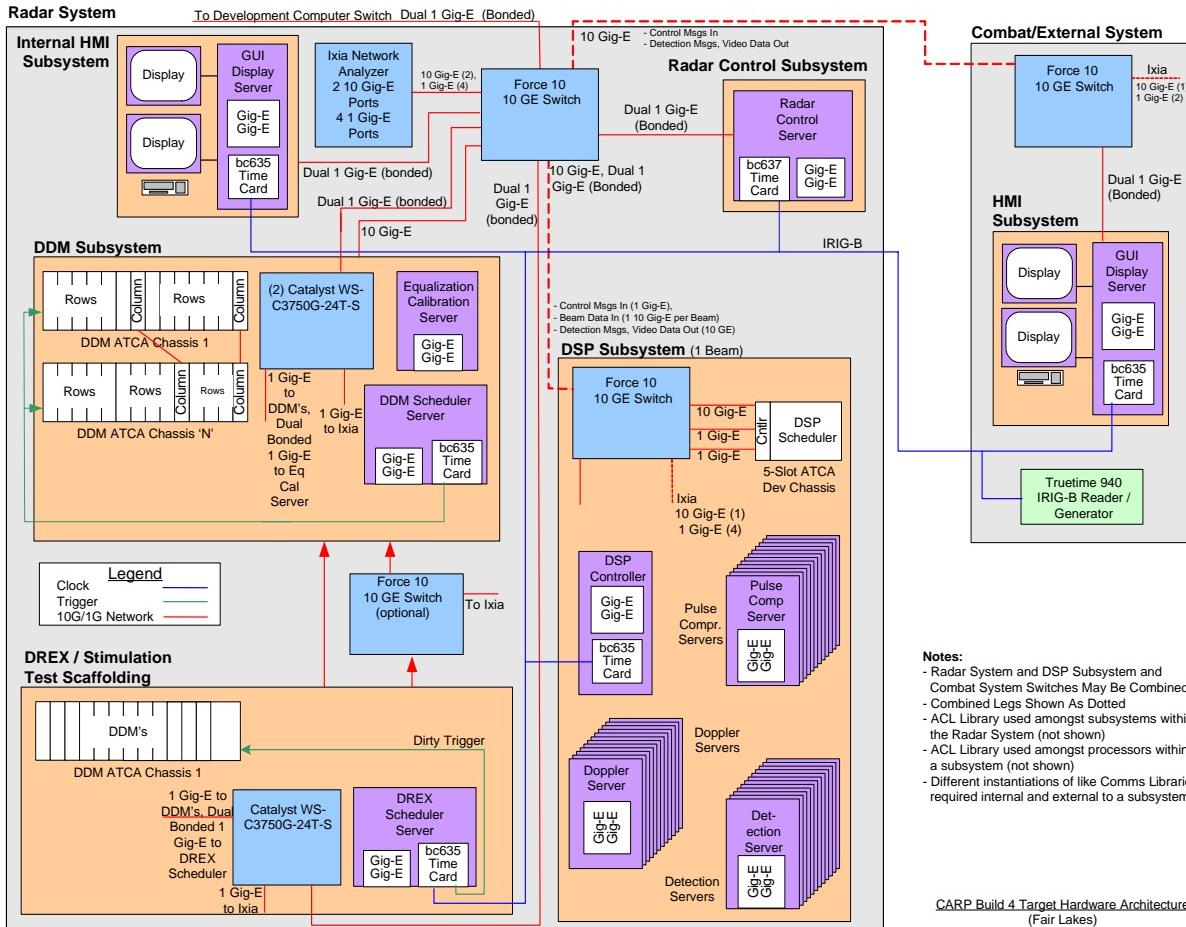
Figure 1: CARP System Conceptual Design

3.5.1 RTAPS Architecture

The three Build 4.1 ROSA-compliant subsystems and the system communications infrastructure are detailed below.

3.5.1.1 DDM Subsystem

The Build 4.1 DDM Subsystem consists of two DDM Beamformer Boards running DDM Controller applications locally using on-board PowerPC processors, and a COTS server-based DDM Scheduler application, providing a subsystem interface to the system (see Figure 2: DDM Subsystem Configuration). The DDM Boards communicate over a high speed Advanced Telecom Computing Architecture (ATCA) Backplane and are connected to the scheduler via copper 1 GE Ports through a 1 GE Network Switch. RF Data input is provided by the DDMs in loopback mode over either internally or over fiber optic 10 GE connections, and beam data output is transmitted out of another fiber 10GE

UNCLASSIFIED

interface and also onto the ATCA backplane. For the purposes of testing, output is captured with an Ixia Network Traffic Analyzer and verified offline with Matlab.
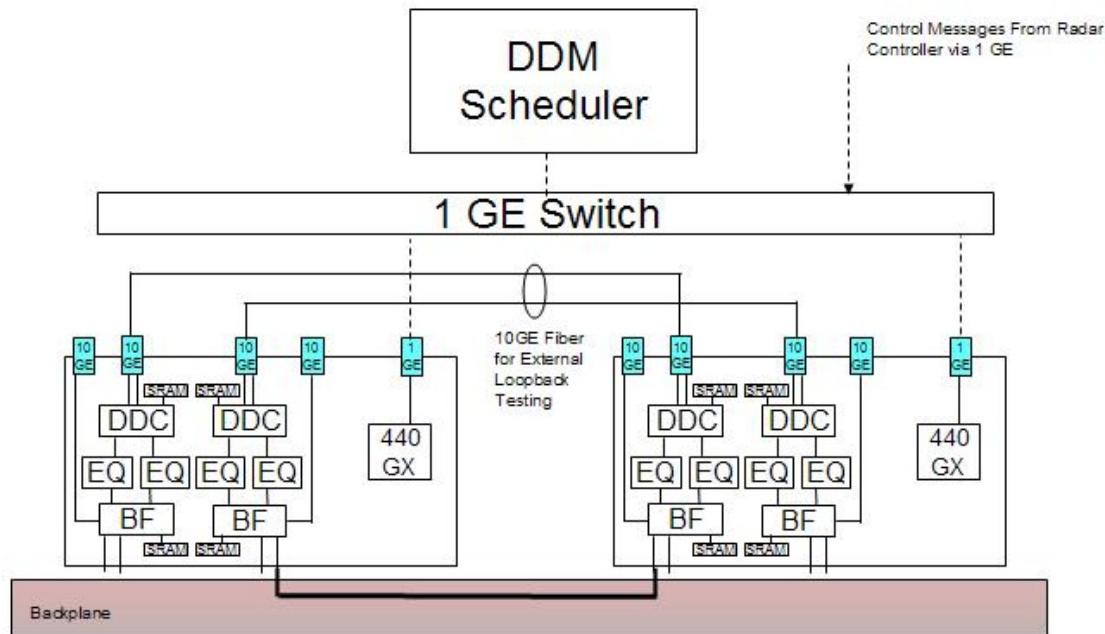
Figure 2: DDM Subsystem Configuration

3.5.1.1.1 Hardware

DDM Subsystem Hardware includes Build 4.1 (rev. 2) DDMs housed in a Schroff Advanced-TCA Chassis, an Apple Xserve G5 Server running a subsystem interface application and a Cisco Catalyst 1 GE Network Switch.

3.5.1.1.1.1   Build 4.1 DDM Board

Similar to Build 3, DDM Boards contain the following components:
- On Board IBM 440GX PowerPC Processor for command and control
    - o 512 Mbytes of SDRAM contained on DIMM module
    - o 96 Mbytes of FLASH Memory
    - o Dual 1 Gigabit Ethernet, selectable between front panel SFP modules and backplane (ATCA base channel)
    - o Serial port to front panel
- Eight FPGAs to perform all channel processing, beamformation, and 10 GE network I/F
- Four FPGAs have local high speed SRAM – 16 Mbytes each
- Four front panel 10GE ports
- Four fabric channels implementing 10 Gigabit Ethernet
- Front and back panel backplane, clocks and triggers

UNCLASSIFIED

- High Speed Low Voltage Differential Signaling (LVDS) signals DDC to EQ, EQ to BF, BF to BF
- Discrete signals between FPGAs for signaling
- PMC slot available for expansion
- Integrated Platform Manager (IPM) per ATCA requirements

Board component functionality is an upgraded version of that delivered for Build 3 (Delivery Order 005, Option 2). The DDM Board has several I/O interfaces on-board to support multiple configurations, including four front panel 10GE ports, four fabric channels supporting 10GE, two front panel 1GE ports, and backplane clocks and triggers on both the front and back panels. Digital Signal Processing on the board is handled by four Xilinx Virtex II Pro P40 and four Xilinx Virtex II Pro P70 FPGAs. DDM dwell to dwell operation is be controlled by the Power PC 440GX, which communicates with the rest of the board hardware via 64/100 PCI-X bus and has access to all board SRAMs (including those local to FPGAs) and a local FLASH memory. The eight FPGAs are interfaced via high speed SPI4_Lite connections. 10 Gbps I/O interfaces run on three devices (XPAC, SFPs, and Fabric IF) and utilize X-Attachment Unit Interface (XAUI) connections. JTAG connectors, a serial port, and front and back panel trigger connectors are included for interfacing with the CARP System.

Hardware upgrades from the Build 3 (rev. 1) board to the current design include:
- Repair of minor design errors in Build 3 Boards
  - Corrected component footprint error
  - Added/changed terminations
  - Removed two soft wires
  - Added a Real Time Clock for general processor use
  - Altered placement of back panel SMA connectors as required by chassis
- Increased processing speed and functionality with improved parts
  - DDC and Beamformer FPGAs increased to -7 speed grade
  - Equalization Filter FPGAs upgraded to P40 with -6 speed grade
  - Increased board clock operating rate from 100MHz to 125 MHz
  - Increased SRAM from 8 MB to 16 MB per DDC and BF FPGA

Design updates on the board were completed by GDAIS in coordination with ADI Engineering, a CARP subcontractor.

UNCLASSIFIED
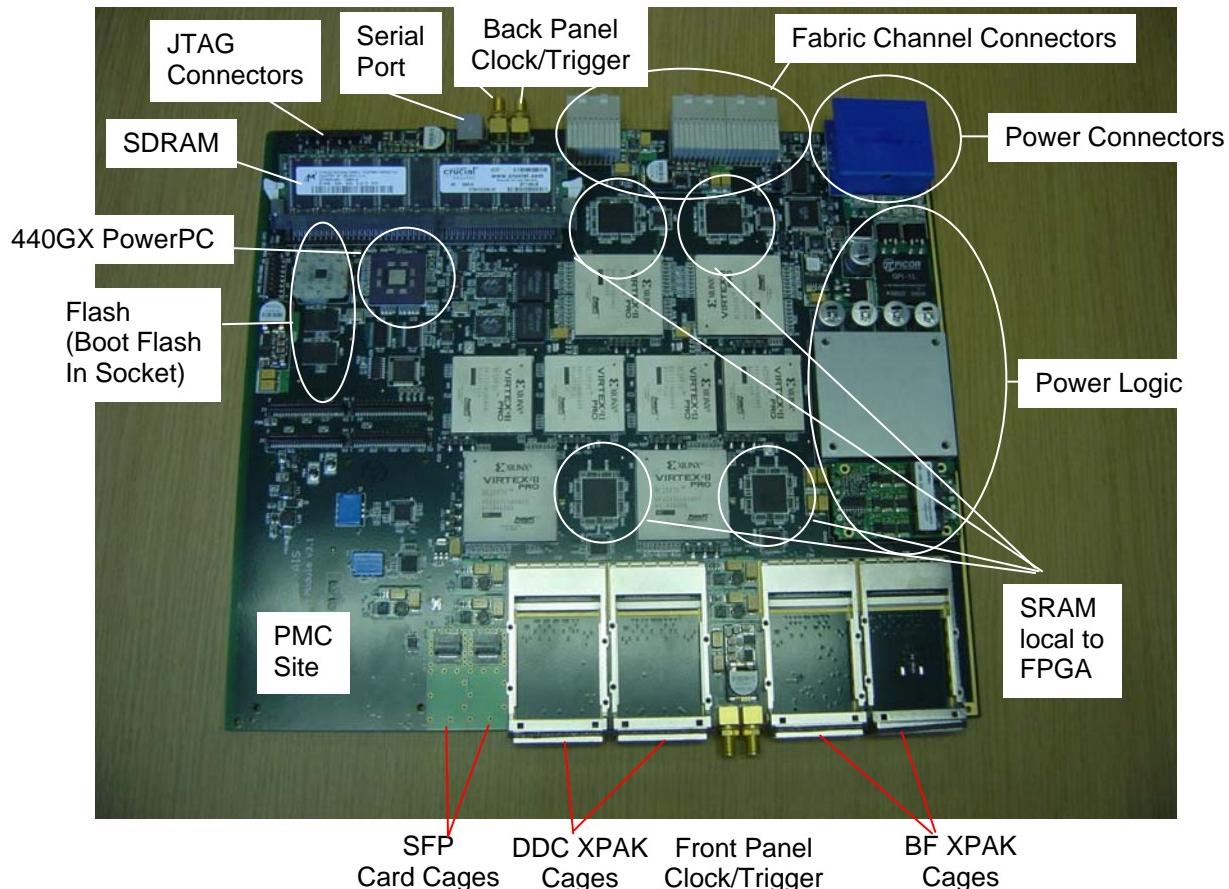
Figure 3: GDAIS/ADI Engineering Data Distribution Module

The block diagram shown in Figure 4 indicates the connections/interfaces of various CARP DDM components.

CAGE CODE	NUMBER	REV
2Y865	1005896	-

7 DECEMBER 2005
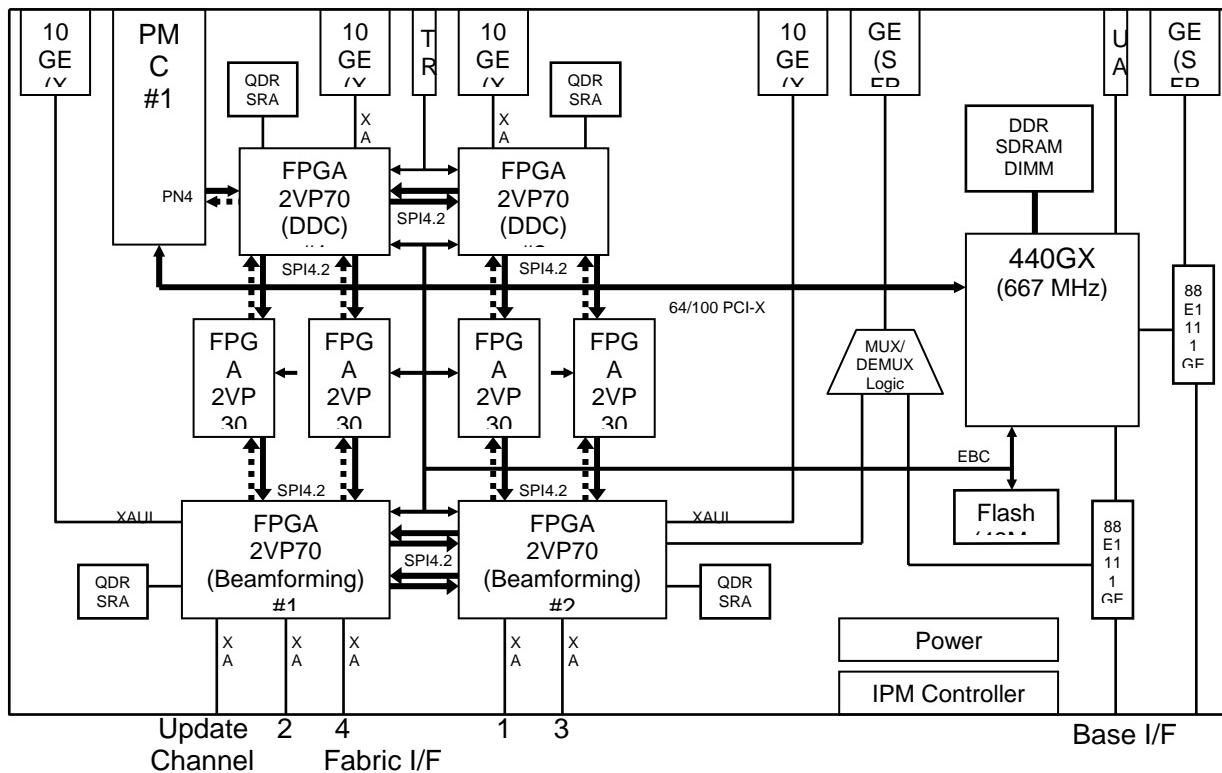
Figure 4: DDM Build 3 Board Configuration

3.5.1.1.1.2   Signal Processing FPGAs

Xilinx FPGAs running VHSIC (Very High Speed Integrated Circuit) Hardware Description Language (VHDL) firmware developed by GDAIS perform three signal processing operations on digital RF signal input: Digital Down Conversion (decimation filtering), channel to channel Equalization, and multi-board row beamforming, and also handle data input, loopback transmissions, and beam data output.

<u>Digital Down Conversion</u>

Digital Down Conversion FPGA Firmware handles DDC Signal Processing and has been upgraded to run on the higher speed grade FPGA (Xilinx part number (Part number XC2VP70-7FF1517C) for Build 4.1. DDC processing is similar to Build 3 and involves band-shifting input signals by mixing with the output of a Numerically Controlled Oscillator (NCO), then filtering and decimation via two polyphase Finite Input Response (FIR) digital filters.

Filter coefficients are stored in FPGA local SRAMs and loaded at startup. Coefficient sets contain 32 16-bit taps and are updatable once per dwell should the configuration change and more then one set be required.

Currently the DDC is implemented with two polyphase filter stages. The first stage performs 2:1 decimation and filtering with an 80% Pass Band, 1.0 dB peak-to-peak ripple, -90 dB out-band rejection and -130 dB image rejection. If utilized, the second adds an additional 2:1 decimation (for a total of 4:1) with the same parameters but a shorter Pass Band (50%).

This component also contains data input loopback transmission logic (detailed below).

Equalization Filtering

Equalization FPGAs perform channel equalization on decimated signals using 32 tap, 16-bit complex FIR filters with coefficient sets provided by the on board processor. The current configuration allows for 1000 sets of updatable coefficients selectable by RX frequency and signal bandwidth. Currently the coefficients are derived offline in a Matlab model and are updateable using communication library tester applications.

The Equalization FPGAs also each include two 16 tap real FIR filters intended to implement true time delay in future builds (one for Inphase samples and one for Quadrature). These filters were implemented and tested but are not in use in the current system as this design (detailed requirements are still being determined in cooperation with the Navy).

Multi-Board Digital Beamforming

The Build 4.1 DDM Subsystem is configured to form four azimuth beams from a row of eight input channels in two DDM boards. Final BF FPGA output product is the result of two processes, partial and full beamforming.

- Each partial beamformer takes accepts two channels of Equalization FPGA output, then multiplies both by appropriate 16 bit complex coefficients and sums
- The FPGAs perform full beamforming by summing their partial beams with other partial Beamformer outputs, either sourcing from the second Beamformer FPGA resident on its board or from the board backplane (issued by the second DDM)

As mentioned, DDM boards interface with each other over an ATCA backplane. Each BF FPGA has two backplane interfaces installed with firmware implemented 10 GE MACs providing a theoretical maximum of 8.0 Gbps between DDMs (due to the current FPCA clock speed). Data destined for transmission over the backplane is additionally routed to front panel 10 GE ports for analysis.

3.5.1.1.1.3   *Board Control*

Board control is provided by the DDM Controller application running on the On Board IBM 440GX PowerPC Processor (see section 3.5.1.1.2 Software).

3.5.1.1.1.4   *Synchronization*

The DDM Subsystem is synchronized to the RTAPS Radar Controller Subsystem by a IRIG-B connection to 1 microsecond. Within the subsystem and across DDM boards, time of day synchronization is maintained by updating local time on reception of one Pulse-Per-Second update signal issued by the DDM Scheduler Server's IRIG-B card.

UNCLASSIFIED

Internally each DDM board maintains time of day with a TOD counter operating at 125 MHz (implying 6 ns resolution). Board local oscillators are specified at 50 Parts Per Million (PPM) (implying a potential clock drift of 50 microseconds per second). Lab measurements taken during subsystem development indicate board-to-board drift of 2.5 microseconds per second relative to each other, and 9 microseconds per second relative to the DDM Scheduler server.

3.5.1.1.1.5   Loopback Stimulation

DDM Boards provide stimulation to the CARP system by issuing pre-generated RF data files stored locally in DDC FPGA SRAMs. The system supports two loopback modes, external and internal.

In external mode, when the DDM Subsystem is configured for loopback stimulation and a radar event is executed, logic in the DDC FPGAs retrieves the appropriate channel data sets from their local SRAM and transmit it over front panel 10 GE fiber optical interfaces to connected DDM input ports for processing.

In internal mode, data files are in the same manner as above to DDC front panel optical interfaces and also internally to DDC FPGA inputs. This mode allows analysis of the loopback channel output without interrupting DDM input.

3.5.1.1.1.6   Data Insertion and Extraction

Firmware installed on all DDM FPGAs provides tapping functionality for extracting data at any DSP stage output to local SRAMs. When the system is configured for data tapping and a radar event is executed, DDM Controller applications retrieve tapped data products from the board SRAMs and save them using the Network File System (NFS) protocol to the subsystem scheduler server.

3.5.1.1.1.7   Scheduler Server

The DDM Scheduler process is resident on an Apple Xserve 1U Server with dual 2.3 GHz PowerPC G5 processors running Yellow Dog Linux 4.0.0 (2.6 Kernel), with 2 GB SDRAM, 1.35 GHz frontside bus, dual on-board 1 GE network interfaces, and a BC635 PCI Card in IRIG-B DCLS Mode for synchronization to an external source (within 1 microsecond) and within the subsystem.

3.5.1.1.2   Software

DDM Subsystem software development and integration was a major goal of Build 4.1, allowing DDM board control and subsystem operation. A real-time Linux-variant operating system was installed on the DDM Boards and, along with an FPGA Device Driver, is utilized by a DDM controller application to manage board operation. As mentioned, a DDM Scheduler process oversees subsystem operation and interfaces with other CARP system components.

DDM Controller

A DDM Controller is the primary application running on DDM board PowerPCs to provide overall control of radar functions. It handles maintenance type commands (such as State/Mode handling, advisory messaging, status messaging, heartbeats, etc.), configures the board for loopback transmissions and data tapping, maintains time-of-day synchronization, and receives and handles radar event messages.

UNCLASSIFIED

FPGA Device Driver

An FPGA device driver was developed to work with a ROSA-compliant FPGA Interface Library (section 3.5.1.4.2.1, Libraries) for accessing and controlling the FPGAs resident on DDM boards. The driver allows usage of DDM FPGAs as a typical Unix device, providing read and write access to FPGA registers and SRAMs and handling interrupts from FPGAs to deliver to the FPGA Interface Library.

Real Time OS

TimeSys Linux is the selected Linux variant operating system installed on DDM boards. Utilizing a Linux variant OS provides a similar environment to other target processors/servers running in the system and, being open source, allows for custom changes as required. TimeSys is a real-time OS supplying the board with the following features:

- Support for high resolution clocks and timers using standard POSIX interfaces
- Support for priority inheritance on mutexes, using the standard POSIX "Threads" library
- Support for periodic tasks, using an Application Programming Interface (API) provided by the TimeSys Linux kernel
- Schedulable interrupt processing
- Schedulable SoftIRQ processing
- Constant time scheduler
- Precise time accounting

DDM Scheduler

The Build 4.1 DDM Scheduler provides buffering and separation between the system commands and controls and those used within the subsystem.

All DDM Subsystem-bound control messages are received by the DDM Scheduler. The scheduler alters and forwards messages to the DDMs as necessary, including inserting data capture fields into received radar event messages prior to sending them to the DDMs. This separation maintains adaptability in the system design and within the subsystem allowing changes in either without impacting the other.

All status, advisory and acknowledgment messages from the DDM subsystem are sourced through the DDM Scheduler. By collecting messages and status reports from individual DDMs, the scheduler avoids inundating external systems with redundant messages and maintains anonymity and a higher degree of autonomy/modularity of individual subsystem components in the system.

The scheduler also interfaces with a timing card to provide the subsystem with synchronization to the rest of the system (see section 3.5.1.1.1.4, Synchronization) and coordinates functions not considered "system" functions to the subsystem, including synchronization, and time-of-day measurements and adjustments. It coordinates subsystem component state and mode transitions, and in the future will manage subsystem calibration.

3.5.1.2   HMI Subsystem

The Build 4.1 HMI Subsystem provides the operator interface to the RTAPS and includes the CARP HMI application running on a PowerPC Server. The HMI application is an updated version of the one used in CARP Build 3, with enhancements to support new system capabilities and added functionality.

UNCLASSIFIED

ROSA-compliant status, control, and radar event messages are exchanged between the HMI and the rest of the system via CARP communication libraries and MTM over the system COTS 1GE Network infrastructure. The subsystem is synchronized to the CARP Radar Controller by Network Time Protocol (NTP) in this build.

3.5.1.2.1 Hardware

HMI hardware components include:
- Apple Power Mac G5 Tower Chassis
  - Dual 2.3 GHz PowerPC G5 Processors Running Yellow Dog Linux 4.0.91 (2.6 Kernel)
  - 2 GB SDRAM, 1.35 GHz Frontside Bus
  - Dual 1 Gigabit PCI Ethernet Cards, Dual Head PCI ATI Radeon Graphics Card

3.5.1.2.2 Software

The Build 4.1 HMI Application is an updated version of the application used in Build 3. It is intended to function as a development and test tool, as well as provide a more automated means of operational control. The Build 4.1 application includes the following windows and displays:
- A Manual-Dwell window for issuing radar event requests
  - The operator specifies radar dwells for the system to execute by selecting from available waveform parameters and entering event parameters manually
- Subsystem Capabilities displays provide access to capability messages issued from the subsystems of the RTAPS that specify system component configurations
- Subsystem detailed status displays are used to request and view subsystem status messages
  - The main HMI display also includes a "rolled-up" status indicator, which lists subsystems being monitored by the HMI along with a color coded indication of their current status, using a standard status level green/yellow/red summary color codes
- An advisory list is provided to display advisory messages issued by RTAPS subsystems
  - Advisories indicate what subsystem component sent them and what event triggered them with an ID number and a short text explanation. They are color coded by severity level in the display (the system advisory severity levels are: debug, info, warning and error).
- An advisory control window provides an advisory message filter to the operator, allowing selection of minimum reported level for each subsystem in the system

3.5.1.3 RC Subsystem

The Build 4.1 RC Subsystem provides Radar Control to the CARP system (similar to RC in previous builds); its main functions are system resource management and radar event scheduling. It is also the Time-of-Day source for the rest of the system. The Build 4.1 RC application runs on a PowerPC G5 server.

3.5.1.3.1 Hardware

RC hardware components include:
- Apple Xserve G5 1U Chassis
  - Dual 2.3 GHz PowerPC G5 Processors Running Yellow Dog Linux 4.0.0 (2.6 Kernel)

UNCLASSIFIED

- o 2 GB SDRAM, 1.35 GHz Frontside Bus
- o Dual On-Board 1 Gigabit Ethernet
- o BC635 in IRIG-B DCLS Mode

3.5.1.3.2   Software

The Build 4.1 RC application is based on the version delivered in Build 3, with enhanced system control functionality and radar event scheduling.

Its main purpose is to provide radar event scheduling to the RTAPS using received radar event requests and build Radar Event Messages to issue to the DDM subsystem for processing. The application also maintains the system Time-of-Day using an integrated Symmetricom bc635 timing card and sources time to the RTAPS in IRIG-B and NTP formats.

3.5.1.4   System Infrastructure

The subsystems of the RTAPS communicate via CARP Communication Libraries running GDAIS MTM across COTS 1 GE network switch configurations.

3.5.1.4.1   Hardware

Infrastructural hardware components include:
- Cisco Catalyst WS-C37506-24T-S 1 GE Switch (2, stacked)

3.5.1.4.2   Software

The following section details CARP communication libraries operating in the RTAPS.

3.5.1.4.2.1   Libraries

CARP Build 4.1 development included upgrading and expanding communication, management and special purpose libraries.

Communications Libraries

Upgraded CARP communications libraries allow RTAPS architecture inter process communication. The following are Build 4.1 Communications Libraries:
- Application Communication Library (ACL): Provides an API to the system for inter-application communication and control, as well as status and capabilities messaging
- Network Resource Allocation Library: Provides data transmission functions, intended for use in DSP applications
- Radar Control Library (RCL): Supplies the Build 4.1 system with radar control functions
- RF Resource Allocation Library (RFRAL): Provides management of RF resource allocation
- RF Resource Control Library (RFRCL): Supplies API calls for controlling RF resources
- GPS Interface (GPSIF): Accesses the GPS timing cards
- FPGA Interface Library: Contains functions for operating and interfacing with DDM FPGAs

UNCLASSIFIED

Library usage is diagramed in Figure 5, below.
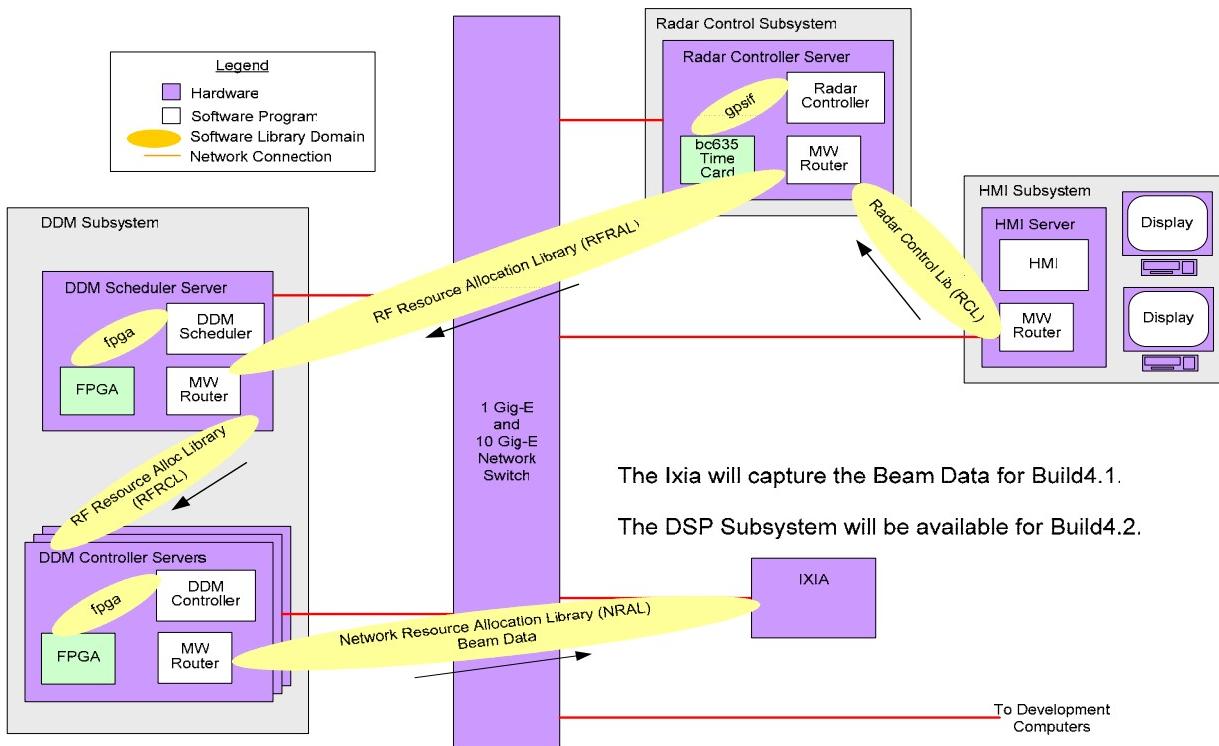
Figure 5: RTAPS Inter Process Communications

Support Libraries

RTAPS utilizes support libraries to assist various programs and systems:
- Middleware Library (MW): Supplies the system with calls to utilize MTM
- Linked List Library (LL): Provides Linked List calls to the system
- Parameter Input Library (PIN): Provides program input routines to the system
- Threads Library: Supports process threads
- Utility I/O Library (UTIO): Provides I/O routines
- C++ Utility Library (CPPU): Utilities used by applications built with C++
- Human Machine Interface Library (HMI): Supplies function calls required by the HMI

Library Test programs were provided for every communication library to verify library and application functionality and allow direct user access to library API calls.

UNCLASSIFIED

3.5.2   Radar Event Message Flow

The Build 4.1 System processes Radar Events requested at the HMI Subsystem via DDM Subsystem Loopback stimulation. The system message flow for processing a Radar Event through the subsystems defined above is as follows:

1) Manual Dwell requests are specified using the CARP HMI (looping functionality allows the definition of thousands of dwells)
2) The HMI Subsystem issues ROSA-compliant Radar Event requests to the RC Subsystem/Application
3) RC Schedules dwells, grouping multiple events into single Radar Event Messages and issues them to the DDM Subsystem
4) The DDM Subsystem Scheduler verifies event parameters and event time, then determines whether or not to capture data during execution.
5) The scheduler then sends the radar events (containing multiple dwell commands) to the DDMs
6) DDM Controllers receive the event messages and perform basic parameter verification, then calls FPGA Interface Library API functions to handle events, which place them in an event queue
7) When FPGAs indicate they are available, the FPGA Interface Library initializes registers in the FPGAs in preparation for the next event
8) FPGAs indicate when registers are available by issuing an interrupt on completion of each radar event
9) The library sets up FPGA Time-of-Day registers with the start time of the next event, then wait for the next interrupt
10) At the appropriate time, the FPGAs will start the radar event, then interrupts the processor to indicate they are ready for setup for the next event
11) If any processing errors are detected, interrupts will be issued to the processor indicating the problems. On detection of a problem, the FPGA Interface Library will perform corrective actions as appropriate, and issue advisory messages to the scheduler for forwarding to the system

3.6   Integration and Test

Testing of the RTAPS was completed after subsystem testing and integration of the three subsystems to validate operation and processing in all components. Along with specific vender provided hardware functional tests (not detailed here), system functionality was verified, data processing was validated, subsystems were exercised independently and integrated with the system, and operational performance was measured.

3.6.1   Data Processing Validation

Verification of system data products was performed throughout development and testing to ensure processing occurred as designed and data message formats were correct. Testing was performed at all development stages from individual component verification through subsystem and system integration and finally in system testing, for all data rates for all supported inputs sizes and types.

Matlab fixed point models were used to generate data sets for stimulation and provided data for comparison and processing validation. All stages of Build 4.1 processing (DDC, Equalization Filtering, and Row Beamforming) have been shown to match the expected results exactly, and meet system goals for Build 4.1. Output formats are as specified by the ICD and are ROSA compliant.

UNCLASSIFIED

3.6.2   Subsystem Functional Testing

Subsystem testing included ensuring each subsystem performs as designed and verifying open architecture-compliance throughout.

Communication Library testers were used to exercise and verify that each subsystem functions as designed and responds correctly to stimulation specified in the ICD. Subsystem internal functional processing was verified using tester and debug tools. The three Build 4.1 subsystems were independently found to be completely functionally operational as designed and ROSA-compliant.

3.6.3   System Functional Testing

System testing involved ensuring that the system works as designed with all designed functionality present and operational.   All subsystems and events execute with appropriate timing and synchronization, and system performance is within Build 4.1 goals. The following sections list a selection of the Build 4.1 System Test Cases.

3.6.3.1   System Test Case 1: HMI Operation and Dwell Scheduling

Objective:
This test was performed to verify that the system handles the processing of Operator Specified Radar Events properly, and schedules requested radar dwells correctly. It involved initializing the system with loopback stimulation files, requesting radar events, capturing output using an Ixia Network Traffic Generator, and monitoring subsystem status during execution using the HMI.

Stimuli:
The stimuli for this test was three pre-generated loopback stimulation files of varying supported lengths (from very small to very large), loaded into the DDM Subsystem configured for external loopback.

Manual-dwell radar event requests were specified using the HMI window, requesting repetitions of dwells for each of the files in varying order. During execution of the events, row beam data output was captured using the Ixia for verification later with Matlab. The HMI was used to check subsystem status messages to ensure all events were being scheduled and executed properly. The ability to terminate dwell processes was also exercised.

Results:
The integrated system functioned as designed from end-to-end, with the HMI Subsystem generating radar event requests from user input, the RC Subsystem scheduling dwells and issuing radar event command messages, and the DDM Subsystem providing external loopback stimulation and properly processing data. Beam data generated during test execution was captured, compared to expected results generated by Matlab fixed-point models and found to match exactly. Status messages issued to the HMI showed radar event processing at each subsystem was as expected. Operator issued termination requests also were handled properly by the system.

UNCLASSIFIED

3.6.3.2   System Test Case 2: System Monitoring

<u>Objective:</u>
The objective of this test case was to verify messaging capabilities between the subsystems were complete, including status reporting, advisory messaging, and heartbeat transmission and monitoring. It involved stimulating the system with error conditions and monitoring subsequently issued advisories, requesting and verifying status messages, and manipulating heartbeat transmissions from various subsystems to ensure notification was appropriate.

<u>Stimuli:</u>
The RF stimulus for this test was a canned loopback stimulation file with a detectable overflow condition present. Detection of the condition when servicing a manual dwell request resulted in the issuing of an advisory message by the DDM subsystem to be received and displayed to the operator by the HMI. Beam data output for the file was captured using the Ixia and compared to Matlab fixed point model results for verification.

Heartbeat handling was verified by restricting the heartbeat of subsystems by temporarily disconnecting them from the subsystem, verifying that notification is issued, reconnecting, and checking that notification expires with the subsystem's return to the system. Status reporting was checked by requesting status messages for all subsystems from the HMI and validating received reports manually.

<u>Results:</u>
The CARP system correctly monitors system status and heartbeats and returns advisories as designed. The DDM subsystem detected the overflow condition and alerted the operator. Beam data output matched expected results exactly. The loss and return of heartbeats from either the RC or DDM Subsystem was correctly handled by the HMI. The subsystems correctly respond to status requests and the resulting HMI display matches the design.

3.6.3.3   System Test Case 3: Synchronization and Timing

<u>Objective:</u>
This test was performed to verify that high-resolution synchronization is in place and it meets the system goals for each subsystem. As mentioned previously, the RC functions as a time of day source to the system by providing IRIG-B DCLS to the DDM subsystem and NTP to the HMI subsystem for synchronization. The DDM subsystem time of day should be kept within 1 microsecond.

<u>Stimuli:</u>
DDM Subsystem synchronization was verified by using an oscilloscope to compare the one Pulse-Per-Second (PPS) output of its bc635 timing card to the output on the RC Subsystem machine.

NTP synchronization of the HMI was checked using an NTP client application running on the HMI Subsystem machine to ensure the RC was sourcing time correctly.

Results:

Figure 6, below, shows the rising edges of the 1PPS outputs of the two subsystems synchronized by IRIG-B synchronization. Comparison of the curser locations indicates that the DDM Subsystem's time-of-day is within about 310 nanoseconds, well within the 1 microsecond goal.
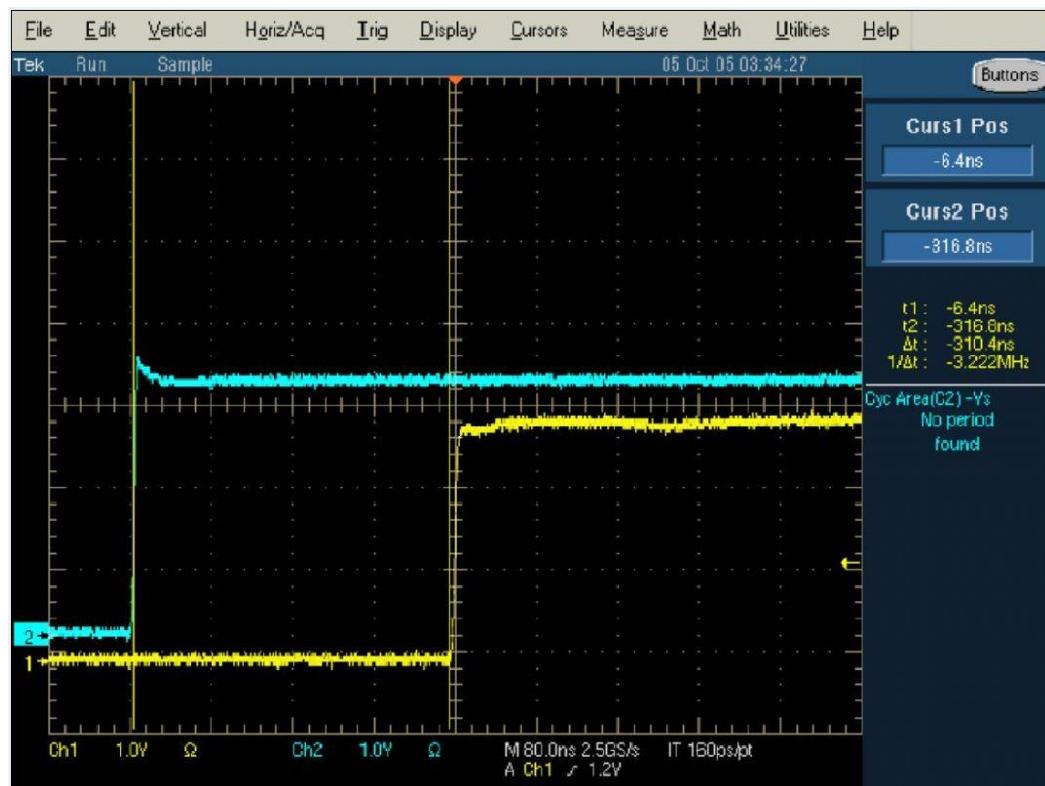

Figure 6: RC - DDM Subsystem Synchronization

NTP synchronization of the HMI to the RC Subsystem was also found to be in place as designed.

3.6.3.4   System Test Case 4: Stimulation Verification

Objective:

The objective of this test was to measure the channel-to-channel transmission start time variance of DDM Subsystem loopback output channels when performing loopback mode stimulation. The DDM Subsystem was designed with a goal of maintaining synchronization in these stimulation channels to within 2.5 microseconds.

Stimuli:

The data set used for this test contained a specific data signature recognizable to a properly configured Ixia Network Traffic Analyzer for latency testing, and was loaded into all channels for loopback

UNCLASSIFIED

stimulation. This Ixia latency measurement setup is capable of measuring the arrival time of captured captures from two stimulating channels with 20 picosecond accuracy.

Results:
Channel-to-channel loopback stimulation variance was found to be well within the 2.5 microsecond goal, with a maximum measured value less then 100 picoseconds.

3.6.3.5   System Test Summary

All functions of the integrated HMI, RC and DDM Subsystems were exercised. Radar event processing was performed and validated using system control and status monitoring capabilities and data products verified appropriately. Data products and processing synchronization were validated using oscilloscopes, network analyzers and Matlab models. The 4.1 System was found to be functionally operational as designed, and meets all synchronization and functional goals.

3.6.4   System Performance Measurements

The following system performance metrics were recorded during testing:

- Radar Event Setup Time: It was found that the average radar event setup time in the Build 4.1 system is between 250 and 500 microseconds depending on data load in the DDM subsystem. Occasionally setup times in the milliseconds are seen. The future goal for this value is to reduce it to less than 100 microseconds.
- Interrupt Handler Processing in Timesys Linux was measured to take between 77 and 195 microseconds.
- Radar Event Advanced Notice: Network throughput and application performance measurements indicate that currently a minimum of 2 ms is required between RC dwell scheduling and event start time is required for radar event processing to occur. In Build 4.1 regular operation, this system is run with 10 ms advance notice built in. The ultimate goal for this value for future builds is under 8 ms.
- Subsystem to Subsystem synchronization via IRIG-B was measured to be within the 1 microsecond goal.
- Channel to Channel loopback transmission synchronization was found to be within the 2.5 microsecond goal.
- Data Rates: The system was shown capable of operating at both supported input rates (100 MSPS and 160 MSPS) and all supported output rates, transmitting beam data at up to 80 MSPS.

3.7   Configuration Management

Build 3 Configuration Management was handled by the Concurrent Versions System (CVS), an open source version control suite. Separate repository trees were maintained for CARP Software, Firmware and Matlab models. More information on CVS is available at www.cvshome.org.

UNCLASSIFIED

CAGE CODE	NUMBER	REV
2Y865	1005896	-

4 FUTURE BUILD DEVELOPMENT PLANS

Build 4 of the Common Affordable Radar Processor is still in process. A full, enhanced, CARP receive system is in development under contract Delivery Order 011 with support for the Digital Array Radar program built in. The final plan is to develop and integrate an antenna subsystem with the goal of live radar system testing.

Specific goals for the CARP project, including those to be completed under future phases of the program contract are:
- Receive System
    - Integrate and demonstrate a scaleable azimuth and elevation processing solution
    - Integrate a server based Digital Signal Processing subsystem
- Upgraded DDM subsystem
    - Target row/column DDMs & digital exciter/receiver simulation DDM configuration
    - Expand BF engine to support 40+ bit Beam Data (I/Q) within DDM subsystem
    - Enhance transmit/stimulation capability
- Interfaces and processing modifications to meet DAR program I/F requirements – e.g.,
    - Continued Open Interface development
    - Support for multiple control schemes
- Design robust Digital Receiver/Exciter (DREX) and Antenna Simulation Subsystems to provide necessary stimulation to the DDM subsystem
- Enhanced DSP
    - Robust detection processing
    - Continuous False Alarm Rate (CFAR) normalization processing
    - Display data pre-processing
- Enhanced Operator Interface Displays

Architecture similar to the concept in Figure 7 is currently in development:
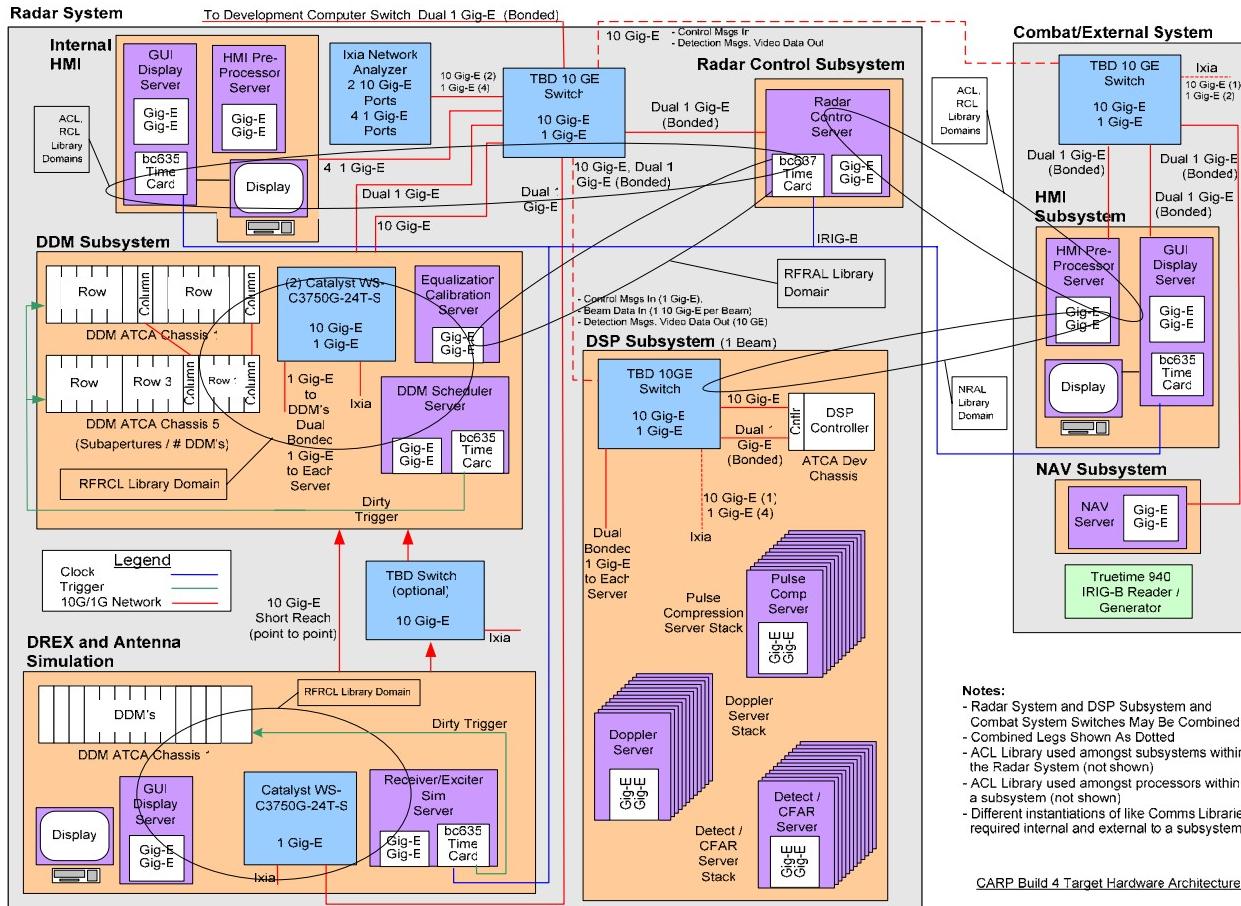
Figure 7: Notional Build 4 Architecture

CAGE CODE	NUMBER	REV
2Y865	1005896	-

5 CONCLUSIONS/LESSONS LEARNED

The CARP team has been able to successfully meet the goals set forth for the program by delivering significant capability on-time.

Under the Build 4.1 effort, the team defined the subsystems of a Radar Open System Architecture Compliant Real-Time Array Processing Subsystem and developed an Interface Control Document for the system. They designed, developed and integrated three subsystems (RC, HMI and DDM) of the RTAPS consistent with the ROSA concept and wrote an ICD for the system. The team has significantly upgraded and enhanced RTAPS functionality in all system functions, and tested and quantified system functionality and performance.

Test results highlight the need to further explore live exercise/stimulation opportunities. The vision for the evolution of this technology is to further develop the antenna subsystem and receiver/exciter prototypes and capabilities for continued development and evaluation of existing digital beamforming, DSP, and associated controls and user interfaces (Figure 8, below).
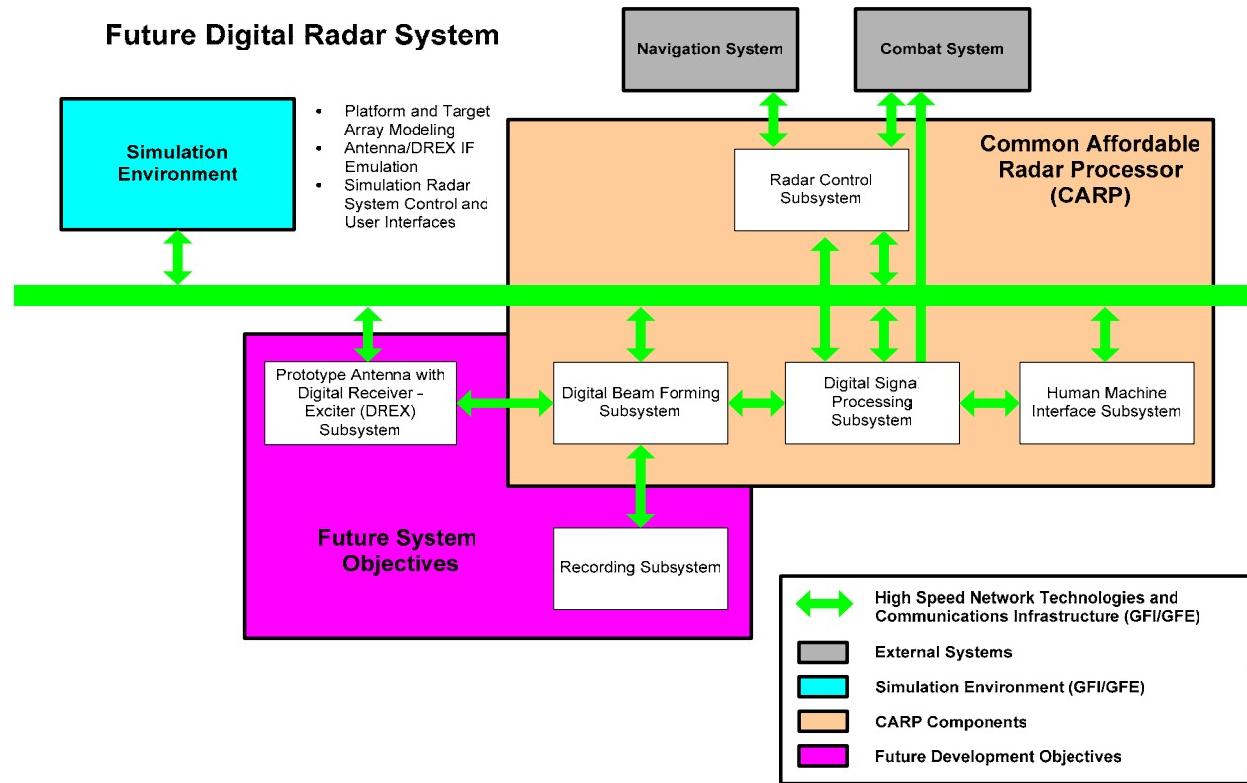

Figure 8: Future Digital Radar System

6   ACRONYMS

ACL	Application Control Library
ADC	Analog to Digital Converter
API	Application Programming Interface
ATCA	Advanced Telecom Computing Architecture
CARP	Common Affordable Radar Processor
CFAR	Continuous False Alarm Rate
COTS	Commercial Off The Shelf
CPLD	Complex Programmable Logic Device
CPPU	C++ Utility Library
CVS	Concurrent Versions System
CW	Continuous Wave
DAC	Digital to Analog Converter
DDC	Digital Down Converter
DDM	Data Distribution
DDMC	DDM Controller
DP	Doppler Processor
DREX	Digital Receiver/Exciter
DSP	Digital Signal Processing
EC	Equalization Calibration
EQ	Equalization
FIFO	First In First Out
FIR	Finite Input Response
FPGA	Field Programmable Gate Array
GPSIF	GPS Interface
HMI	Human Machine Interface
ICD	Interface Control Document
IF	Intermediate Frequency
IPM	Integrated Platform Manager
IQ	Inphase/Quadrature
JTAG	Joint Test Action Group
LFM	Linear Frequency Modulated
LL	Linked List Library

LVDS	Low Voltage Differential Signaling
MTM	Multipurpose Transportable Middleware
MW	Middleware
NCO	Numerically Controlled Oscillator
NFS	Network File System
NIC	Network Interface Card
NRAL	Network Resource Allocation Library
NTL	Nav/Time Library
NTM	Nav/Time Manager
NTP	Network Time Protocol
ONR	Office of Naval Research
PC	Pulse Compressor
PCI	Peripheral Component Interface
PECL	Positive Emitter Coupled Logic
PIN	Parameter Input Library
PMC	PCI Mezzanine Card
PRI	Pulse Repetition Interval
RC	Radar Controller
RCL	Radar Control Library
RFRAL	RF Resource Allocation Library
RFRAM	RF Resource Allocation Manager
RFRCL	RF Resource Control Library
RTAPS	Real Time Array Processing Subsystem
ROSA	Radar Open System Architecture
TOD	Time of Day
UTIO	Utility I/O Library
VHDL	Very High Speed Integrated Circuit (VHSIC) Hardware Description Language
VHSIC	Very High Speed Integrated Circuit
XAUI	X-Attachment Unit Interface

UNCLASSIFIED